

Assignment #10

Due **Monday 8 December at 5:00pm** ← revised!

Please read Lectures 20, 21, 22, 23, 24, 25, 26, and 27 in the textbook *Numerical Linear Algebra*, SIAM Press 1997, by Trefethen and Bau.

DO THE FOLLOWING EXERCISES FROM THE TEXTBOOK:

- **Exercise 21.1** *This shows how Matlab's `det` is actually implemented, in $O(m^3)$ work. Recall that expansion in minors implies more than $O(m!)$ work.*
- **Exercise 22.1** *Hint. See the last line of Algorithm 21.1. What do you know about ℓ_{jk} that allows you to generate an upper bound?*
- **Exercise 24.1**

DO THE FOLLOWING ADDITIONAL PROBLEMS:

P24. (a) Implement Algorithm 26.1, Householder reduction to Hessenberg form. Specifically, build a code with the signature

`H = hessen(A, stages)`

Your code will check that A is square, and return a Hessenberg matrix H such that $A = QHQ^*$ for some unitary Q . It will print the stages if `stages` is `true`. For simplicity, your code can discard the vectors v_k after they are used.

(b) For a random 5×5 matrix A of your choice, run the code and show the four stages A , $Q_1^*AQ_1$, $Q_2^*Q_1^*AQ_1Q_2$, and $H = Q_3^*Q_2^*Q_1^*AQ_1Q_2Q_3$, where $H = \text{hessen}(A)$. (Hint. Thereby show the the matrix patterns in the “A Good Idea” subsection on pages 197–198.) Use the built-in `eig()` command to show that the eigenvalues of A and H are the same to within rounding error.

(c) Construct a new 7×7 Hermitian matrix S and compute $T = \text{hessen}(S)$. Check that T is tridiagonal and Hermitian. Again use `eig()` to show that the eigenvalues of S and T are the same within rounding error.

P25. Over the course of the semester, several times we have used a two-word phrase (2WP) to describe a matrix factorization. There are 10 possible 2WPs which can be formed from a column-A adjective and a column-B noun from the following table:

<i>column A</i>	<i>column B</i>
triangular	triangularization
orthogonal or unitary	orthogonalization
	diagonalization
	tridiagonalization
	hessenbergization

Consider only the textbook Lectures which we covered, specifically Lectures 1–17 and 20–27. For square matrices, which distinct factorizations use, or could use, one of these 10 possible 2WPs? For each such factorization, state the matrix factorization, its 2WP, its usual/official name, and reference(s) to the textbook. Reference Algorithms in the textbook too, if any.

Hints. The adjectives “orthogonal” and “unitary” applied to matrices are essentially synonyms, and I suggest you treat them that way. However, in listing your factorizations you need to see that there are different ways to apply orthogonal/unitary matrices, so make sure you consider all types. Regarding your final list, you can get full credit without perfect agreement with my list, but not if you miss important factorization ideas.

(Extra Credit 1) The SVD factorization is *not* one of the factorizations above. Why? Invent a good 2WP to describe it.

(Extra Credit 2) Develop and implement triangular hessenbergization.

P26. This exercise “discovers” Fourier series while considering eigenvalues/vectors of certain matrices. The matrix $F \in \mathbb{C}^{m \times m}$ with entries $f_{jk} = (f_k)_j$, from formula (2), is the discrete Fourier transform.

A circulant matrix is one where constant diagonals “wrap around”:

$$(1) \quad C = \begin{bmatrix} c_1 & c_m & \cdots & c_3 & c_2 \\ c_2 & c_1 & c_m & & c_3 \\ \vdots & c_2 & c_1 & \ddots & \vdots \\ c_{m-1} & & \ddots & \ddots & c_m \\ c_m & c_{m-1} & \cdots & c_2 & c_1 \end{bmatrix}$$

Each entry of $C \in \mathbb{C}^{m \times m}$ is determined from the entries c_1, \dots, c_m in its first column:

$$C_{jk} = \begin{cases} c_{j-k+1}, & j \geq k, \\ c_{m+j-k+1}, & j < k. \end{cases}$$

Specifying the first column of a circulant matrix describes it completely.

An extraordinary fact about a circulant matrix is that it has a complete set of complex eigenvectors *that are known in advance*, without knowing the eigenvalues. Specifically, define $f_k \in \mathbb{C}^m$ by

$$(2) \quad (f_k)_j = \exp\left(-i(j-1)(k-1)\frac{2\pi}{m}\right) = e^{-i2\pi(k-1)(j-1)/m},$$

where, as usual, $i = \sqrt{-1}$. These vectors are *waves*, that is, combinations of familiar sines and cosines. After some warm-up exercises you will show in part (e) that $Cf_k = \lambda_k f_k$ for a computable eigenvalue λ_k .

- (a) Define the *periodic convolution* $u * w \in \mathbb{C}^m$ of vectors $u, w \in \mathbb{C}^m$ by

$$(u * w)_j = \sum_{k=1}^m u_{\mu(j,k)} w_k \quad \text{where} \quad \mu(j,k) = \begin{cases} j - k + 1, & j \geq k, \\ m + j - k + 1, & j < k. \end{cases}$$

Show that $u * w = w * u$.

- (b) Show that $Cu = v * u$ if C is a circulant matrix and v is the first column of C .
- (c) Show that the vectors f_1, \dots, f_m defined in (2) are orthogonal. (*Hints. Remember the conjugate in the inner product. Then consider finite geometric series.*)
- (d) For $m = 20$, use Matlab to plot the real parts of the vectors f_1, \dots, f_5 , together in a single figure. (*Hint. They look like discretized waves.*)
- (e) For a general circulant matrix, C in (1) above, give a formula for its eigenvalues λ_k in terms of the entries c_1, \dots, c_m . That is, show via by-hand calculation that

$$Cf_k = \lambda_k f_k$$

where f_k is given by (2). Your solution will contain a formula for λ_k .